Optimization-based parameter estimation of dynamic power grid systems

Cosmin G. Petra

Mathematics and Computer Science Division

Argonne National Laboratory (ANL)

Joint with: Noemi Petra (UC Merced), Zheng Zhang (MIT), Emil Constantinescu (ANL), Mihai Anitescu (ANL)

SIAM – UQ Conference

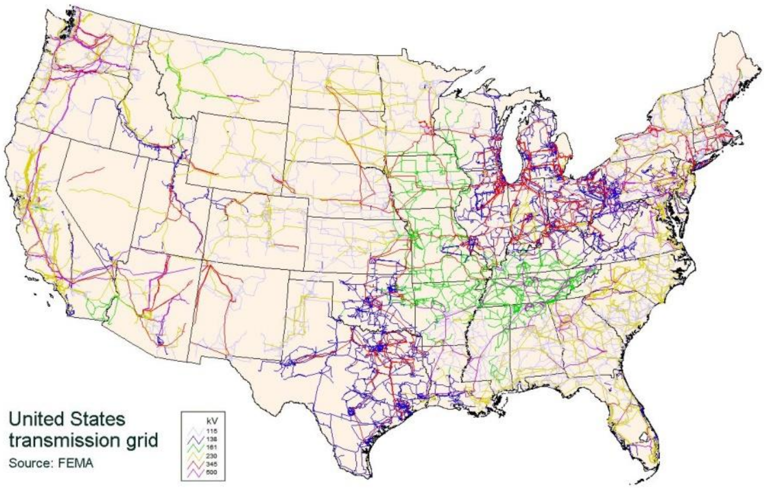
April 8 - 2016

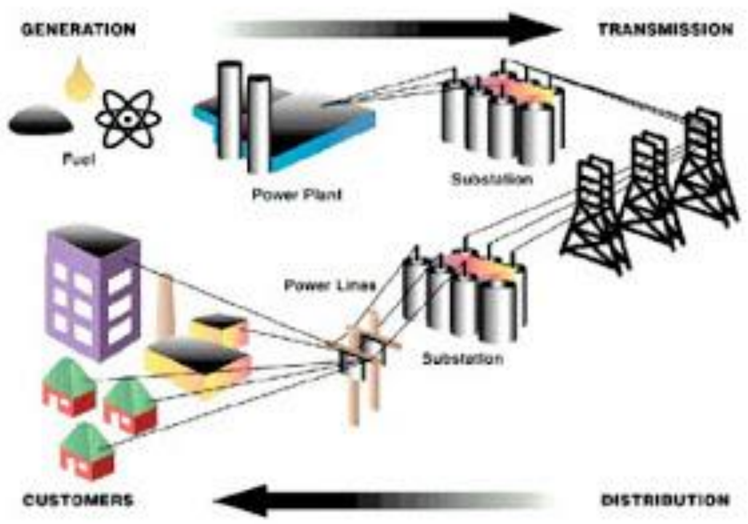
Outline

- **Scalable stochastic optimization**
  - PIPS: high-performance computing for stochastic optimization
  - Application to economic dispatch of electricity in the U.S. power grid
- **Parameter estimation**
  - Application: estimating the inertia of electrical generators using phasor measurements units
  - Problem formulation and computational setup
- **Solution methods and HPC**: dynamic optimization
  - Bonus application: dynamically secure optimal power flow
  - Structured quasi-Newton

Mathematical optimization procedures for power grid

The U.S. power grid is one of the most complex engineering systems.





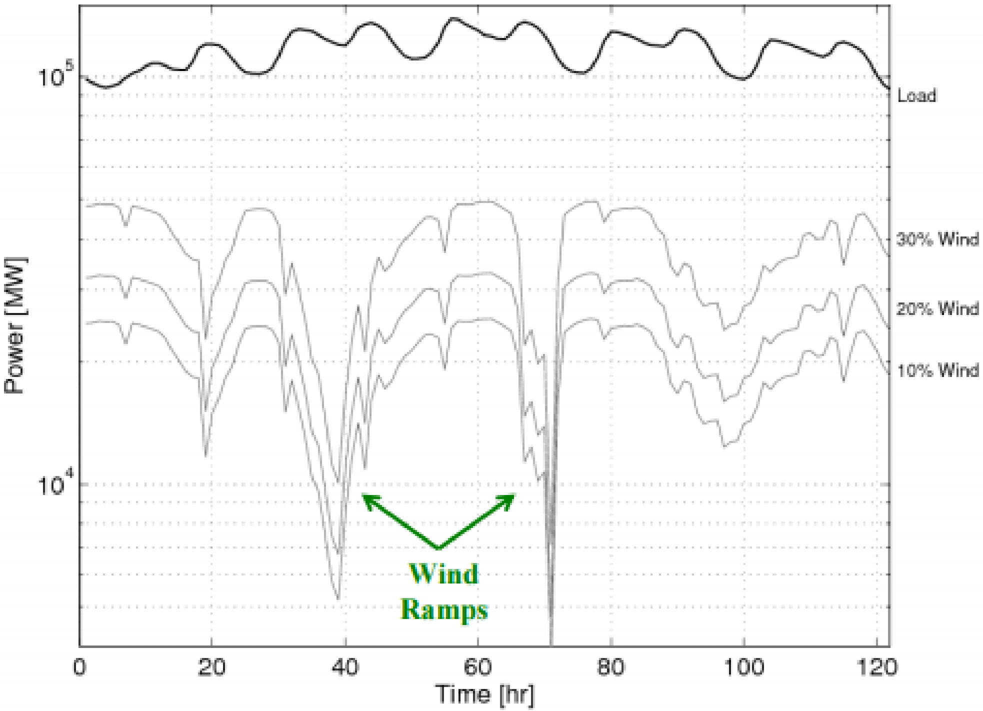
Economic optimization of generation → Stochastic optimization

Monitoring (parameter/state estimation) → Dynamic optimization (defined by dynamical systems)

Optimal control to ensure grid stability → Dynamic optimization (defined by dynamical systems)

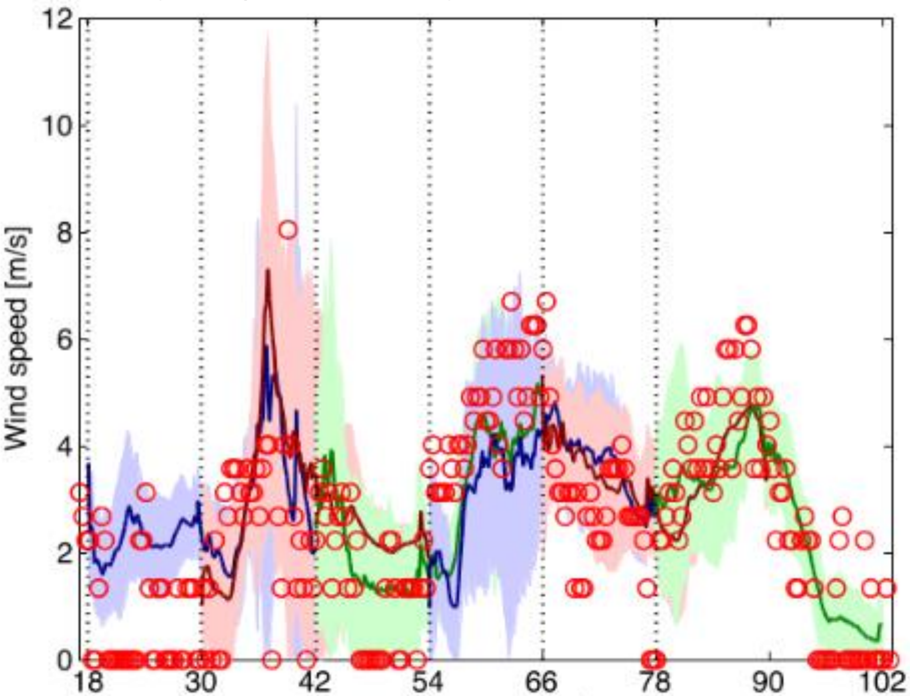
Stochastic optimization on high-performance computers

Electricity generation and dispatch under uncertainty



The sharp drops in wind power need to be forecasted well in advance to give the thermal generators enough time to ramp up generation.

Wind forecasting results in wind scenarios, requiring stochastic optimization



Hours since June 1, 2006, 12am

Stochastic programming

- Two-stage stochastic programming with recourse

$$\underset{x_0}{Min}\left\{ f_0(x_0) + \mathbb{E}_\omega\left[\underset{x}{Min}\ f(x,\omega) \right]\right\}$$

subj. to. $A_0 x_0 = b_0$ subj. to. $\mathrm{W}(\omega)x = b(\omega) - T(\omega)x_0$

$x_0 \geq 0$ $x \geq 0$

- Stochastic linear programming: $f_0(x_0) = c_0^T x_0,\quad f(x,\omega) = c(\omega)^T x$

$\xi(\omega) := (\mathrm{W}(\omega), \mathrm{T}(\omega), b(\omega), c(\omega))$ → **Sampling** → $\xi_1, \xi_2, \ldots, \xi_N$

- **Sample average approximation (SAA)**

$$\underset{x_0, x_1, x_2, \ldots, x_N}{Min}\quad f_0(x_0) + \frac{1}{N}\sum_{i=1}^{N} f_i(x_i)$$

$$\begin{array}{llcl} \text{subj. to.} & A_0 x_0 & = & b_0 \\ & T_k x_0 + \quad W_k x_k & = & b_k, \\ & x_0 \geq 0, \quad x_k \geq 0, & & k = 1, \ldots, \mathrm{N}. \end{array}$$

Parallel interior-point methods for stochastic optimization

$$
\begin{array}{rcccccccccccc}
\min & c_0^T x_0 & + & c_1^T x_1 & + & c_2^T x_2 & + & \dots & + & c_N^T x_N & & \\
\text{s.t.} & A x_0 & & & & & & & & & = & b_0, \\
 & T_1 x_0 & + & W_1 x_1 & & & & & & & = & b_1, \\
 & T_2 x_0 & & & + & W_2 x_2 & & & & & = & b_2, \\
 & \vdots & & & & & & \ddots & & & & \vdots \\
 & T_N x_0 & & & & & & & + & W_N x_N & = & b_N, \\
 & \multicolumn{11}{l}{x_0 \ge 0, \quad x_1 \ge 0, \quad x_2 \ge 0, \quad \dots, \quad x_N \ge 0.}
\end{array}
$$

$$
\begin{bmatrix}
K_1 & & & B_1 \\
 & \ddots & & \vdots \\
 & & K_N & B_N \\
B_1^T & \dots & B_N^T & K_0
\end{bmatrix}
\begin{bmatrix}
\Delta z_1 \\ \vdots \\ \Delta z_N \\ \Delta z_0
\end{bmatrix}
=
\begin{bmatrix}
r_1 \\ \vdots \\ r_N \\ r_0
\end{bmatrix}
$$

Structured linear systems enable parallel linear algebra

Block Elimination

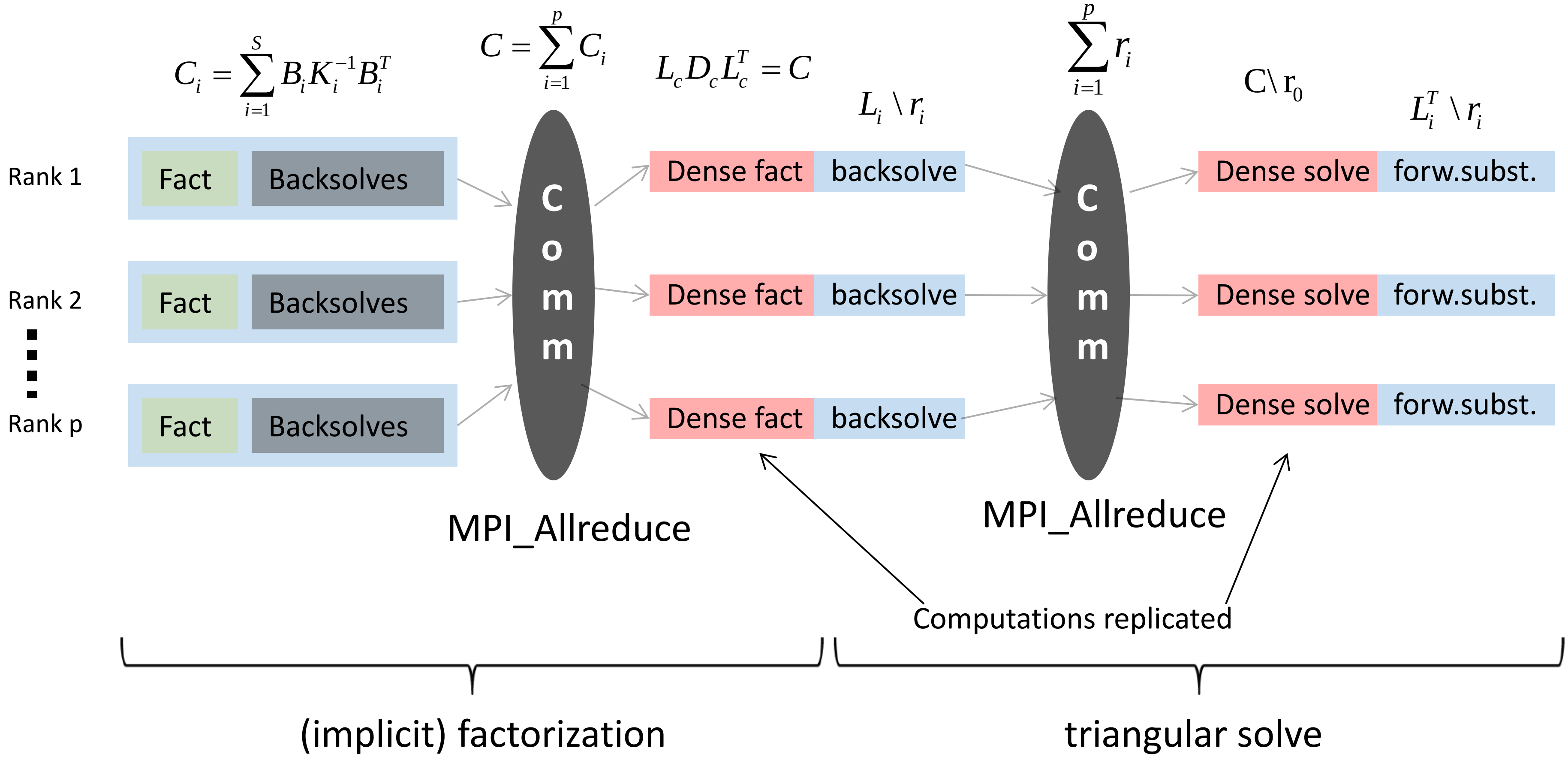
$$\begin{bmatrix} K_1 & & & B_1 \\ & \ddots & & \vdots \\ & & K_N & B_N \\ B_1^T & \dots & B_N^T & K_0 \end{bmatrix} \begin{bmatrix} \Delta z_1 \\ \vdots \\ \Delta z_N \\ \Delta z_0 \end{bmatrix} = \begin{bmatrix} r_1 \\ \vdots \\ r_N \\ r_0 \end{bmatrix}$$

Multiply row *i* by $-B_i^T K_i^{-1}$ and sum all the rows to obtain

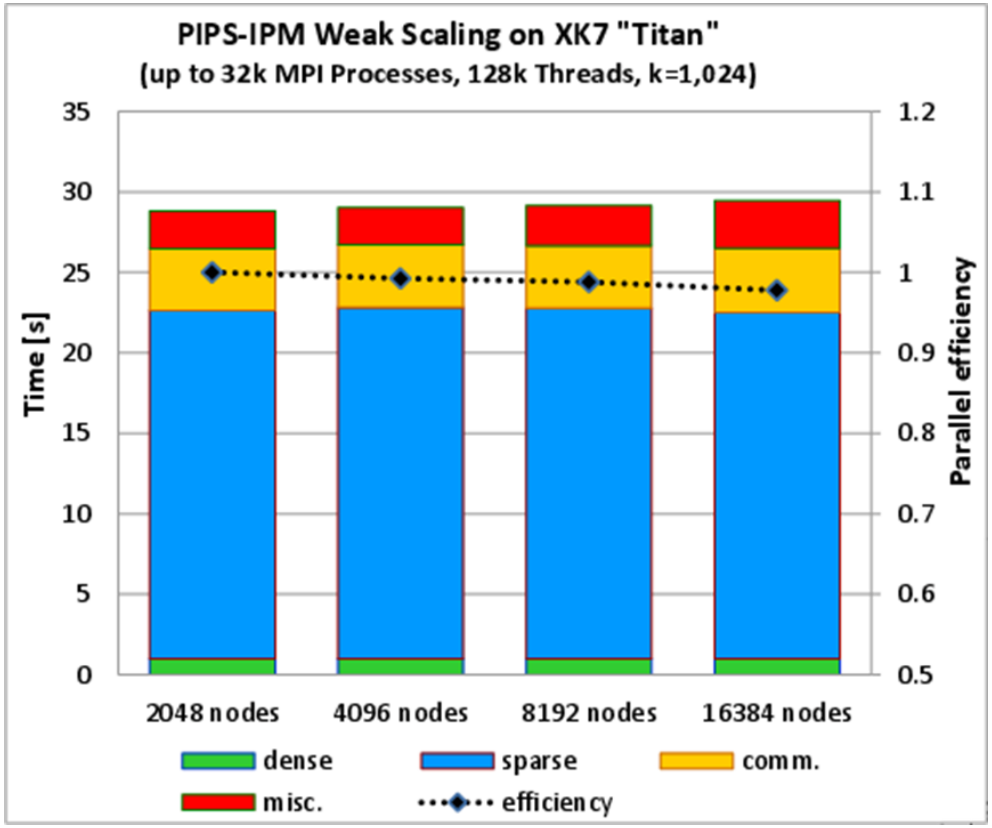
$$\left(K_0 - \sum_{i=1}^{N} B_i^T K_i^{-1} B_i \right) \Delta z_0 = r_0 - \sum_{i=1}^{N} B_i^T K_i^{-1} r_i$$

The matrix $C := K_0 - \sum_{i=1}^{N} B_i^T K_i^{-1} B_i$ is the Schur-complement of the diagonal $K_1, \dots, K_N$ block.

Parallel computational pattern



Efficiency of block-angular linear algebra

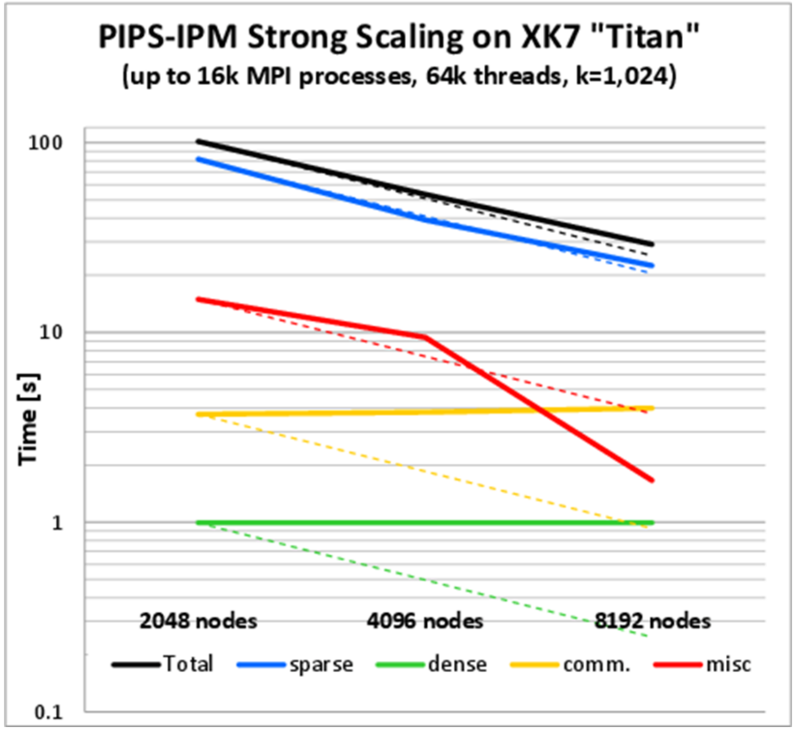


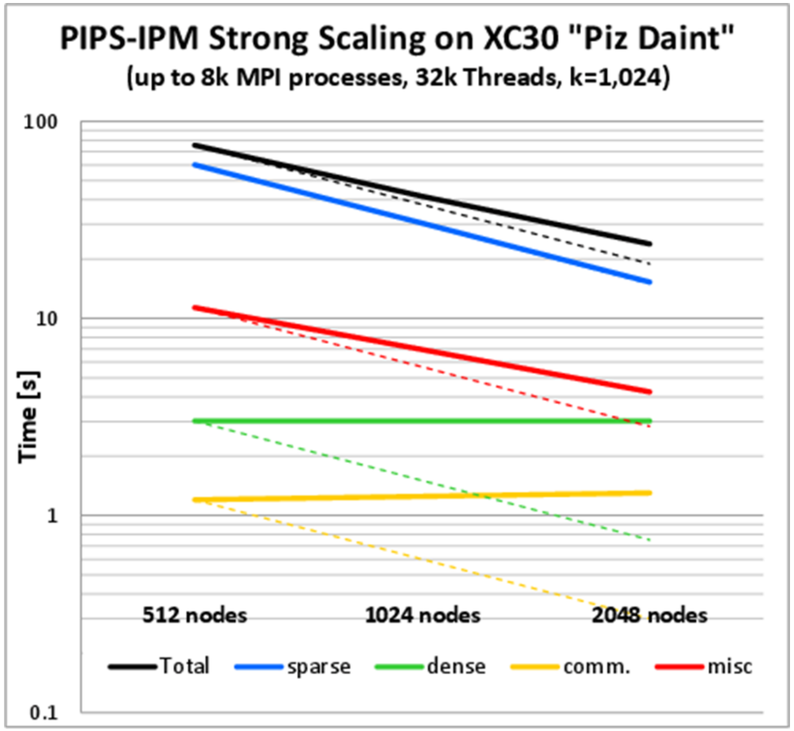
Economic dispatch for the State of Illinois with up to 32,768 scenarios

Largest instance has 4.08 billion decision variables and 4.12 billion constraints (32,768 scenarios).

Execution time is approx. constant each time the size of the problem and the number of nodes is doubled.

Strong scaling – optimization and computational components





The Illinois ED instance used in the XK7 runs has 4.08 billion decision variables and 4.12 billion constraints (32,768 scenarios).

Optimization parallel efficiency

Solve to completion - Illinois ED on Intrepid BG/P

Nodes/scens	Wall time (sec)	IPM Iterations	Time per IPM iteration (sec)
4096	3548.5	103	33.57
8192	3883.7	112	34.67
16384	4208.8	123	34.80
32768	4781.7	133	35.95

About 75% efficiency.

The algorithm is also scalable

PIPS-IPM/NLP

- Open-source, available at **https://github.com/Argonne-National-Laboratory/PIPS**
- Development started in 2009
  - Other contributors: N. Chiang, M. Lubin, V. Zavala, M. Anitescu
- Currently used by Exxon and United Technologies.
- Ported and runs on various HPC platforms, IBM BG/P & Q (ANL), Cray XK7 (ORNL), Cray XC30 and XK7 (Swiss National Computing Centre)
- PIPS-S – parallel revised simplex (Lubin, Petra, Hall (2012))
  - Coin-Or Cup 2013 and COAP Best paper 2013
- Under development: PIPS-SBB - parallel branch bound for mixed integer stochastic programing (Rajan, Obxberry (LLNL), and Petra (ANL))

Dynamic optimization

Inertia estimation during dynamic transients

Joint work with: Noemi Petra (UC Merced), Zheng Zhang (MIT), Emil Constantinescu (ANL), and Mihai Anitescu (ANL)

Transient stability optimal power flow

Joint work with: Naiyuan Chiang (UTRC), Shri Abyankhar,(ANL) and Mihai Anitescu (ANL)

Power grid parameter and state estimation

- Estimation is essential in operating the system: real-time monitoring and fault detection, dynamic stability analysis, transmission switching, etc.

- Data acquisition devices
  - SCADA (supervisory control and data acquisition) – 10 seconds sampling/response rate
  - PMU (phasor measurement unit) – 30 ms sampling rate, measures **dynamic** states (*e.g.,* phase, amplitude)

- Realtime estimation of large power grid systems under live data streaming.

Parameter estimation as a dynamic optimization problem

Mathematical model			Data
Parameters	**States (forward model)**	**Observables**	**PMU observations**
Inertia	**Angle, frequency, etc. (generators) phase, amplitude (buses)**	**Phase, amplitude**	
m	$\begin{aligned} \dot{x} &= h(x,y,m,t) \\ 0 &= g(x,y,m,t) \\ x(0) &= x_0(m) \end{aligned}$	$\mathcal{O} = \text{filter}(x,y)$	d

Parameter-to-observable mapping

$$f : m \xmapsto{\mathcal{F}} (x,y) \xmapsto{\mathcal{O}} \text{observables}$$

Find the parameters that reconcile observables with measurements

$$\min_{m} \quad \mathcal{J}(m) := \frac{1}{2} \| f(m) - d \|^2_{\Sigma^{-1}_{noise}}$$

Power system dynamics described by a DAE system

Generator equations

$$
\begin{aligned}
T'_{doi}\frac{dE'_{qi}}{dt} &= -E'_{qi} - (X_{di} - X'_{di})I_{di} + E_{fdi}, \\
T'_{qoi}\frac{dE'_{di}}{dt} &= -E'_{di} - (X_{qi} - X'_{qi})I_{qi} \\
\delta_i &= \omega - \omega_s \\
\frac{2H_i}{\omega_s}\frac{d\omega_i}{dt} &= T_{Mi} - E'_{di}I_{di} - (X'_{qi} - X'_{di})I_{di}I_{qi} - D_i(\omega_i - \omega_s) \\
T_{Ei}\frac{dE_{fdi}}{dt} &= -(K_{Ei} + S_{Ei}(E_{fdi}))E_{fdi} + V_{Ri} \\
T_{Fi}\frac{dR_{fi}}{dt} &= -R_{fi} + \frac{K_{Fi}}{T_{Fi}}E_{fdi} \\
T_{Ai}\frac{dV_{Ri}}{dt} &= -V_{Ri} + K_{Ai}R_{fi} - \frac{K_{Ai}K_{Fi}}{T_{Fi}}E_{fdi} + K_{Ai}(V_{refi} - V_i) \\
0 &= E'_{di} - V_d - R_sI_{di} + X'_{qi}I_{qi} \\
0 &= E'_{qi} - V_q - R_sI_{qi} - X'_{di}I_{di}
\end{aligned}
$$

Swing equation: generator inertia to be estimated

(7+2 eqns per generator)

Current balance

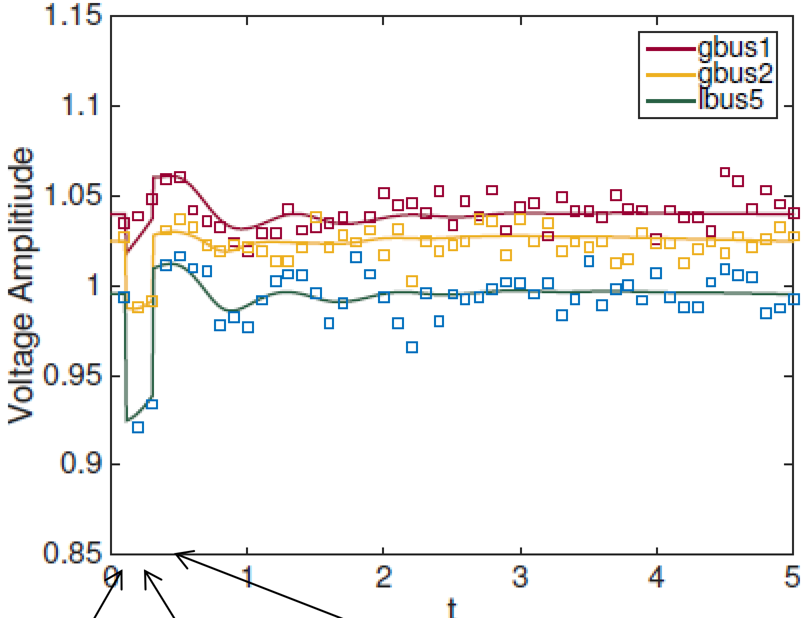
$$0 = I_{gen} - YV - I_{load}$$

(2 eqns per bus)

P. Sauer and M. A. Pai. – “Power Systems Dynamics and Stability”, Stipes Publishing Co, 2007

The inertia estimation problem

- Estimating the generators inertia during a dynamic transient ("bump" in load or drop in generation)
- PMU measurements: voltage phase and amplitude

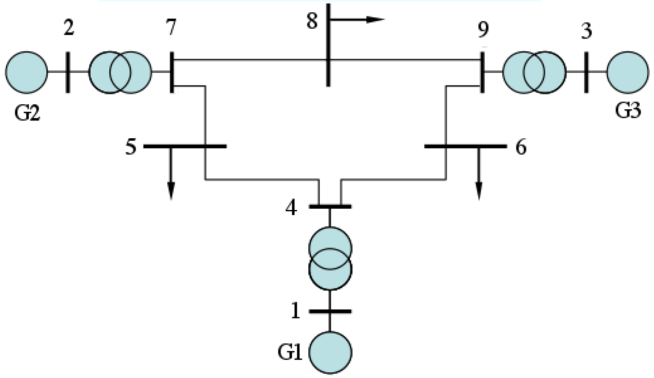


t=[0,0.1]
(steady state)

t=[0.1,0.3]
(load disturbance)

t>0.3
(post-disturbance)

3-gen, 9-bus system



N. Petra, C. G. Petra, Z. Zhang, E. Constantinescu, M. Anitescu, A Bayesian Approach for Parameter Estimation with Uncertainty for Dynamic Power Systems, submitted to IEEE Transactions on Power Systems.

Transient stability optimal power flow (TSOPF)

- **Optimal power flow**: economic optimization (nonlinear economic dispatch)

$$
\begin{aligned}
\min \quad & C(p) \\
\text{s.t.} \quad & g_s(p) = 0 \\
& h_s(p) \leq 0 \\
& p^- \leq p \leq p^+
\end{aligned}
$$

- **Transient stability** studies of the **power system stability** following a major transient (loss of transmission lines, generators, and/or drastic changes in load)

- **TSOPF controls the generators to ensure transient stability at minimum generation cost (steady-state).**

- Example: frequency stability using a penalty approach

$$\underline{\omega} = 59.2\text{Hz} \leq \omega(t,p) \leq \bar{\omega} = 60.8\text{Hz}, \forall t \in [t_0, t_f] \iff H(p) = 0$$

$$H(p) = \int_{t_0}^{t_f} \left[(\omega(t,p) - \underline{\omega})_- + (\omega(t,p) - \bar{\omega})_+\right]^2 dt = 0$$

$$a_- = \max(-a, 0) \text{ and } a_+ = \max(0, a)$$

Computational framework for dynamic optimization

$$
\begin{aligned}
\min \quad & C(p) \\
\text{s.t.} \quad & g_s(p) = 0 \\
& h_s(p) \leq 0 \\
& p^- \leq p \leq p^+ \\
& H_c(p) = 0, \quad c \in \mathcal{C}
\end{aligned}
$$

PIPS structure exploiting for HPC

Algebraically (closed-form) available
First and second order derivatives readily available (ADor hand coding)
(Structured) AD in Julia -> StructJuMP

Simulation-based with adjoint sensitivities

$$H(p) = \int_{t_0}^{t_f} h(x(t,p), y(t,p))dt$$

(one DAE solve)

$$
\begin{aligned}
M\dot{x} &= f(t,x,y,p), \\
0 &= g(t,x,y,p), \\
x(t_0) &= I_{x_0}(p), \\
y(t_0) &= I_{y_0}(p)
\end{aligned}
$$

$$\nabla H(p) = \int_{t_0}^{t_f} f_p^T \lambda dt - ((Mx_p)^T \lambda)_{t=0}$$

(one adjoint solve)

$$
\begin{aligned}
M^T\dot{\lambda} &= -f_x^T\lambda + g_x^T\mu - h_x, \\
0 &= -f_y^T\lambda + g_y\mu - h_y, \\
M^T\lambda(t_f) &= 0, \\
\mu(t_f) &= 0
\end{aligned}
$$

PETSc time-stepping solver and adjoints

Structured secant update

$$\min_{x \in \mathbb{R}^n} \quad f(x) := k(x) + u(x)$$

Available problem information

$$k(x), \quad \nabla k(x), \quad \nabla^2 k(x)$$
$$u(x), \quad \nabla u(x)$$

- Want a symmetric Hessian approximation that uses the available Hessian, namely

$$B_{i+1} = \nabla^2 k(x_{i+1}) + A_{i+1}, \text{ where } A_{i+1} \approx \nabla^2 u(x_{i+1})$$

- Structured BFGS update (derived analytically)

$$A_{i+1}^{BFGS} = A_i - \frac{(A_i + K(x_{i+1}))s_i s_i^T (A_i + K(x_{i+1}))}{s_i^T (A_i + K(x_{i+1}))s_i} + \bar{\gamma}_i \bar{y}_i \bar{y}_i^T, \quad \bar{\gamma}_i = \frac{1}{(K(x_{i+1})s_i + \bar{y})^T s_i}$$

- Convergence in a (modified) line-search framework using Wolfe conditions
- Super linear convergence under standard assumptions
- Provably better performance for low-rank unknown Hessians

Simulation results for inertia estimation problem

t_f	H_1	H_2	H_3	#iter
	$\Delta_t = 0.01,\ \Delta_t^{obs} = 0.02$			
5	23.64	6.41	2.99	19
1	23.62	6.43	2.97	14
0.8	23.59	6.45	2.98	10
0.6	23.60	6.44	3.00	12
0.5	23.87	6.36	3.04	12
0.4	23.88	6.43	2.98	13

Δ_t^{obs}	N_{obs}	H_1	H_2	H_3	#iter
		$t_f = 0.6,\ \Delta_t = 0.01$			
0.01	60	23.69	6.39	3.01	10
0.05	30	23.60	6.44	3.00	12
0.1	6	23.59	6.41	3.04	13
0.5	2	23.20	6.34	3.53	9

Table: A study of the effect of the time horizon (left) and frequency of observations (right) on the ability to recover the inertia parameter. “True” values of the inertias are (23.60, 6.42, and 3.02).

Uncertainty quantification of the inertia estimation

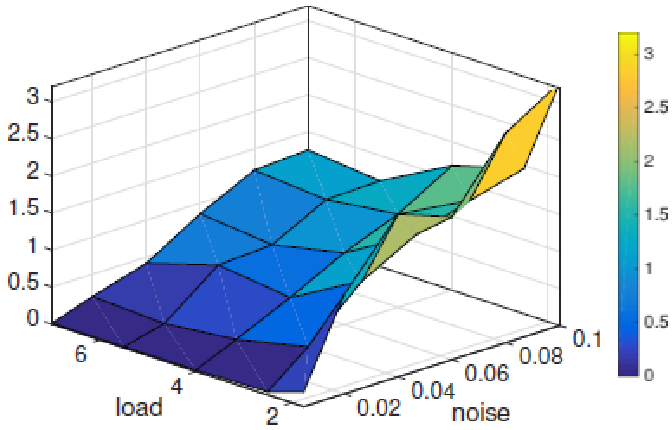


Figure 1: Surface plot of the trace of the Gaussianized posterior covariance as a function of noise and load

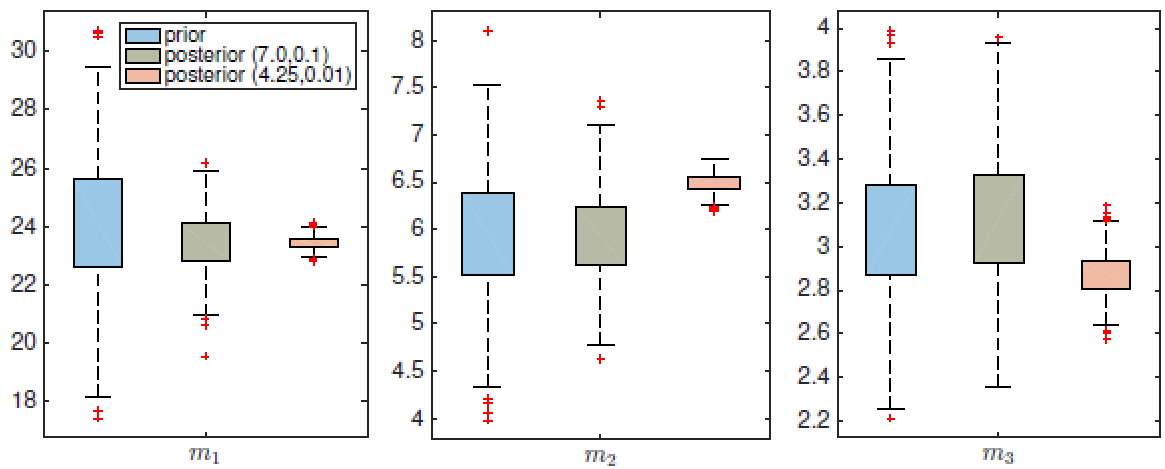


Figure 2: A "whiskers boxplot" of the prior and posterior mean and variances for two values of load and data noise for the three inertia parameters. The central mark is the median, the edges of the box are the 25th and 75th percentiles and the "whiskers" extend to the most extreme data points.

Summary

- Stochastic optimization is a scalable computational paradigm.
- Scalable optimization algorithms are good candidates for solving estimation and optimal control problems.
- General mathematical and computational setup
  - driven by power grid applications but general to accommodate other applications as well.

Future research

- Larger, realistic power grid systems and HPC computing
- Full posterior exploration in high dimensions
- State estimation
  - Incomplete observations: only 20% of the buses are instrumented with PMUs
  - Noisy measurements

Thanks for your attention!

Questions?

Additional references

Petra et al., "*Scalable Stochastic Optimization of Complex Energy Systems,*" in proceedings of Supercomputing'11, 2011.

Petra et al., "*A preconditioning technique for Schur complement systems arising in stochastic optimization,*" Journal of Computational Optimization and Applications, 2012.

Lubin et al., "*On the parallel solution of dense saddle-point linear systems arising in stochastic programming,*" Journal of Optimization Methods and Software, 2012.

Chiang et al., "*Structured Nonconvex Optimization of Large-Scale Energy Systems Using PIPS-NLP,*" 18th IEEE Power Systems Computations Conference, 2014.

Petra et al., "*Real-time Stochastic Optimization of Complex Energy Systems on High Performance Computers,*" IEEE Computing in Science & Engineering (CiSE), 2014

Petra et al., "*An augmented incomplete factorization approach for computing the Schur complement in stochastic optimization,*" SIAM Journal on Scientific Computing, 2014

PIPS – parallel solvers suite for structured optimization

Interface
- Low level: C, C++, and file
- AMPL (hackishly parallel)
- Julia (**JuMP**)
- **StructJuMP** (fully parallel)

Problem formulation
- LP/QP
- Stochastic LP/QP
- Stoch+Network
- Stoch+Network+PDAE
- NLP
- Stochastic NLP
- Dynamic parameter estimation
- Stoch+DAE/ODE Control

Optimization algorithm
- **Mehrotra IPM**
- **Filter IPM**
- **Quasi-Newton IPM**

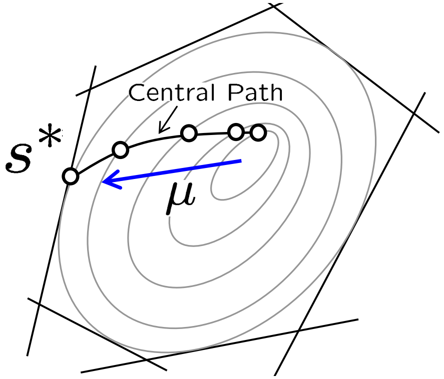
Linear algebra
- block-angular (MPI+X)
- serial (OOQP)
- network partitioning
- Discretized PDAE

OOQP - Object-oriented software for quadratic programming (Michael Gertz and Steve Wright , Argonne, 2003)

Interior-point methods (IPMs)

$$\min_x \quad c^T x \qquad \text{s.t.: } Ax = b,\ x \geq 0 \quad \Longrightarrow \quad \min_x \quad c^T x - \mu \sum \ln x_i \qquad \text{s.t.: } Ax = b \quad (\text{multiplier } y)$$

Idea: Solve a sequence of log-barrier problems ($\mu \to 0$) using Newton method.



- Best known iteration complexity (polynomial) for LPs
- Mehrotra (1992, predictor-corrector primal-dual)
  - Best practical performance
- Filter line search IPMs for nonlinear programming (NLP)

- Direct matrix factorizations required by the ill-conditioned linear systems

$$\begin{bmatrix} \mu X^{-2} & A^T \\ A & 0 \end{bmatrix} \begin{bmatrix} \Delta x \\ \Delta y \end{bmatrix} = \begin{bmatrix} r_x \\ r_y \end{bmatrix}$$

Example: stochastic economic dispatch

Large-scale (dual) block-angular linear programming problem

$$
\begin{array}{llllllllllll}
\min & c_0^T x_0 & + & c_1^T x_1 & + & c_2^T x_2 & + & \dots & + & c_N^T x_N & & \\
\text{s.t.} & A x_0 & & & & & & & & & = & b_0, \\
 & T_1 x_0 & + & W_1 x_1 & & & & & & & = & b_1, \\
 & T_2 x_0 & & & + & W_2 x_2 & & & & & = & b_2, \\
 & \vdots & & & & & & \ddots & & & & \vdots \\
 & T_N x_0 & & & & & & & + & W_N x_N & = & b_N, \\
 & x_0 \geq 0, & & x_1 \geq 0, & & x_2 \geq 0, & & \dots, & & x_N \geq 0. & &
\end{array}
$$

- **Sparse matrices** due to network structure, time coupling, etc.
- Many scenarios (1,000s, 10,000s …) to accurately model uncertainty
- “Large” scenarios (W_i up to 250,000 x 250,000)
- “Large” 1st stage (10,000s, 100,000s of variables)

Quasi-Newton methods

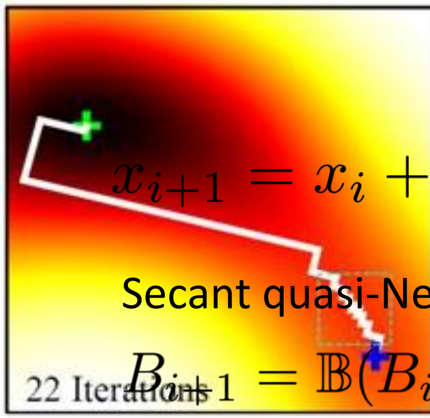
$$\min f(x)$$

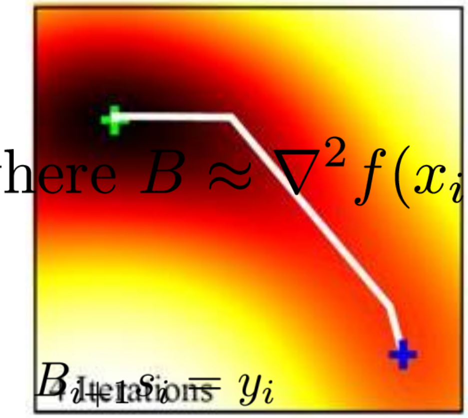
Steepest descent

$$x_{i+1} = x_i + \alpha_i \nabla f(x_i)$$

Newton

$$x_{i+1} = x_i + \alpha_i \left[\nabla^2 f(x_i)\right]^{-1} \nabla f(x_i)$$





Quasi-Newton

$$x_{i+1} = x_i + \alpha_i B^{-1} \nabla f(x_i), \text{ where } B \approx \nabla^2 f(x_i)$$

Secant quasi-Newton approximations

$B_{i+1} = \mathbb{B}(B_i, x_i, s_i, y_i)$ satisfying $B_{i+1} s_i = y_i$

Here $s_i = x_{i+1} - x_i, \; y_i = \nabla f(x_{i+1}) - \nabla f(x_i)$

Unstructured quasi-Newton secant updates

Davidon-Fletcher-Powell (DFP)

$$B_{i+1} = (I - \gamma_i y_i s_i^T) B_i (I - \gamma_i s_i y_i^T) + \gamma_i y_i y_i^T$$

Broyden, Fletcher, Goldfarb, and Shanno (BFGS)

$$B_{i+1} = B_i - \frac{B_i s_i s_i^T B_i}{s_i^T B_i s_i} + \gamma_i y_i y_i^T$$

Deriving the structured BFGS update

$$\begin{array}{ll} \min\limits_{A} & \|(K(x_{i+1}) + A)^{-1} - (K(x_{i+1}) + A_i)^{-1}\|_{F,W} \\ \text{s.t.} & A = A^T,\ As_i = \bar{y}_i \end{array}$$

Notation: $K(x) = \nabla^2 k(x)$

We solved it analytically to obtain the **structured BFGS update:**

$$A_{i+1}^{BFGS} = A_i - \frac{(A_i + K(x_{i+1}))s_i s_i^T (A_i + K(x_{i+1}))}{s_i^T (A_i + K(x_{i+1}))s_i} + \bar{\gamma}_i \bar{y}_i \bar{y}_i^T, \quad \bar{\gamma}_i = \frac{1}{(K(x_{i+1})s_i + \bar{y})^T s_i}$$

Preliminary numerical results

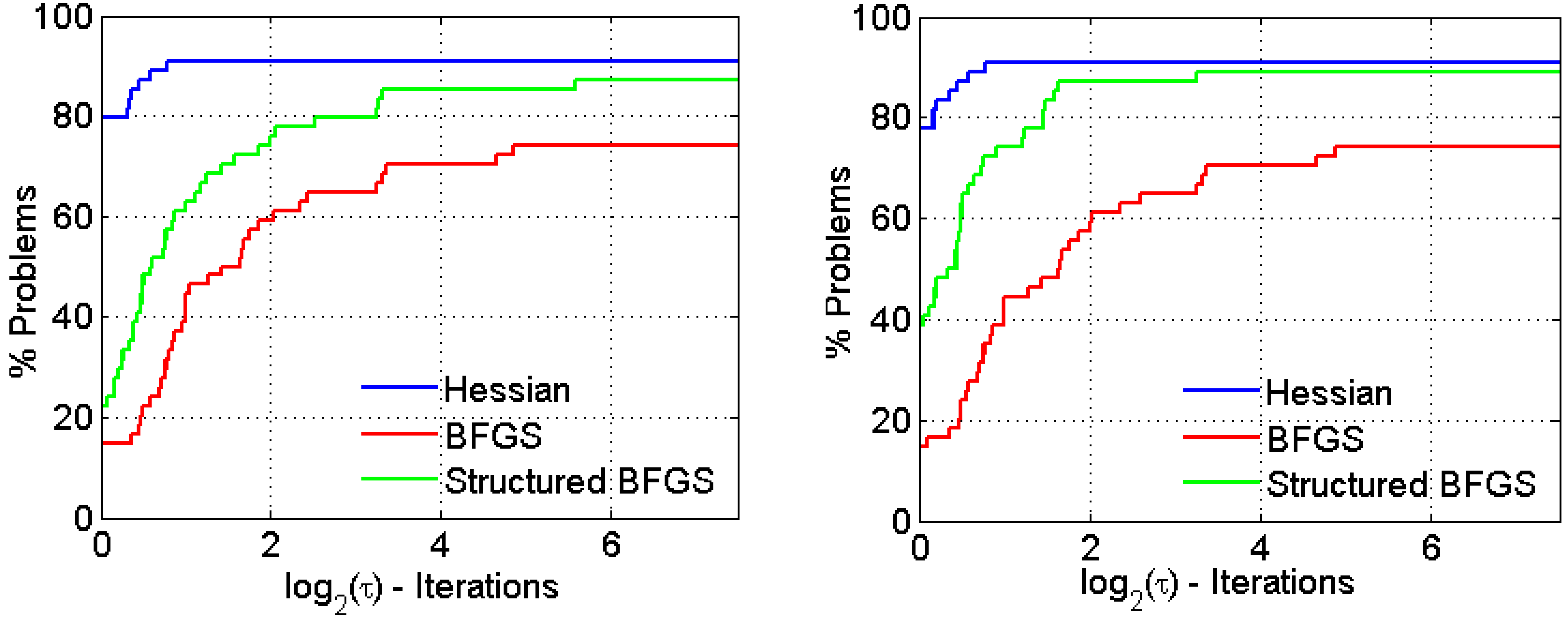


Figure 1: Performance profiles for nonlinear unconstrained CUTEst problems: left plot for problems with full-rank missing Hessian terms, right plot for low-rank missing Hessian terms.

Stochastic optimization in power grid

Minimize generation cost subject to demand and network power flow constraints in the presence of (stochastic) weather conditions → **economic dispatch (ED) / optimal power flow (OPF)**

$$\min_{x,y(\omega),f,F(\omega)} \sum_{i\in G} c_i x_i + \mathbb{E}_\omega \sum_{i\in G} \left(c_i^+ (y_i(\omega) - x_i)_+ - c_i^- (y_i(\omega) - x_i)_-\right)$$

$$\text{subj.to:} \quad \tau_n(f) + \sum_{i\in T(n)} x_i = d_n, \forall n \in N$$

$$\tau_n(F(\omega)) + \sum_{i\in T(n)} y_i(\omega) = d_n, \forall n \in N, \omega \in \Omega$$

$$f, F(\omega) \in U, \forall \omega \in \Omega$$

$$x_i \in C_i, y_i(\omega) \in C_i(x_i, \omega), \forall i \in G, \omega \in \Omega$$

x_i, y_i	-	generation levels	f	-	vector of line flows
c_i	-	generation costs	$\tau_n(f)$	-	net power imported at node n
c_i^+, c_i^-	-	realtime bids (sell and buy)	N	-	set of network nodes
G	-	set of generators (coal, gas, nuclear, wind, solar)	d_n	-	demand at node n
			$T(n)$	-	set of generators at node n
C_i	-	generation constraints	U	-	line capacity constraints

Ω is the set of **weather scenarios**